

REMARKS

Applicant is in receipt of the Office Action mailed July 14, 2004. Claims 167-241, 255-262, 264-282, and 284 have been cancelled. New claims 285-381 have been added to more clearly represent the claimed invention. Thus, claims 285-381 remain pending in the case. Further consideration of the present case is earnestly requested, in light of the following remarks.

Amendments to the claims

Applicant has amended each of the independent claims to clarify that the programmatic creation of the graphical program is performed without any user input selecting graphical program nodes or interconnecting the nodes. In other words, the graphical program is created *automatically*, where graphical program nodes are included in the graphical program and interconnected *without any user input*.

Section 103 Rejections

The Office Action rejected claims 272-282 under 35 U.S.C. 103(a) as being unpatentable over Sojoodi et al (USP 6437805, "Sojoodi 805") and Sojoodi et al (USP 5784275, "Sojoodi 275"). These claims have been cancelled, and so their rejection is rendered moot. Applicant believes that the new claims presented above are patentably distinct and non-obvious over Sojoodi 805 and Sojoodi 275, and provides arguments detailing why below.

As the Examiner is certainly aware, to establish a prima facie obviousness of a claimed invention, all claim limitations must be taught or suggested by the prior art. In re Royka, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974), MPEP 2143.03. Obviousness cannot be established by combining or modifying the teachings of the prior art to produce the claimed invention, absent some teaching or suggestion or incentive to do so. In re Bond, 910 F. 2d 81, 834, 15 USPQ2d 1566, 1568 (Fed. Cir. 1990).

Moreover, as held by the U.S. Court of Appeals for the Federal Circuit in Ecolchem Inc. v. Southern California Edison Co., an obviousness claim that lacks

evidence of a suggestion or motivation for one of skill in the art to combine prior art references to produce the claimed invention is defective as hindsight analysis.

In addition, the showing of a suggestion, teaching, or motivation to combine prior teachings “must be clear and particular Broad conclusory statements regarding the teaching of multiple references, standing alone, are not ‘evidence’.” *In re Dembiczak*, 175 F.3d 994, 50 USPQ2d 1614 (Fed. Cir. 1999). The art must fairly teach or suggest to one to make the specific combination as claimed. That one achieves an improved result by making such a combination is no more than hindsight without an initial suggestion to make the combination.

Claim 370 recites:

370. (New) A system for programmatically creating a graphical program, comprising:

- a computer system including a CPU and memory;
- a client program executing in the computer system, wherein the client program performs API calls to programmatically create a graphical program; and
- a server program operable to receive the client program calls to programmatically create the graphical program, wherein, in programmatically creating the graphical program, the server program is executable to:
 - programmatically create a plurality of graphical program nodes in the new graphical program; and
 - programmatically interconnect the plurality of graphical program nodes in the new graphical program, wherein the interconnected plurality of graphical program nodes comprise at least a portion of the new graphical program;

wherein the server program is operable to programmatically create the new graphical program without any user input specifying selection of graphical program nodes and interconnection of graphical program nodes.

Applicant notes that in asserting motivation for the alleged combination of Sojoodi 805 and Sojoodi 275, the Examiner has simply cited an improved result (“the motivation being to enable the system to wire nodes to produce program logic and data

flow”), which is improper. Applicant respectfully submits that neither Sojoodi 805 nor Sojoodi 275 provides a motivation to combine, and so the references are not properly combinable to establish an obviousness rejection, and that even were Sojoodi 805 and Sojoodi 275 properly combinable, which Applicant argues they are not, the resulting combination would still not produce Applicant’s invention as claimed.

For example, Applicant notes that the cited column 5, lines 28-30 of Sojoodi 805 states “The present invention comprises a system and method for creating a graphical program, wherein the graphical program is operable to access capabilities of an object”, but does not describe *server program is executable to: programmatically create a plurality of graphical program nodes in the new graphical program; and programmatically interconnect the plurality of graphical program nodes in the new graphical program, ... wherein the server program is operable to programmatically create the new graphical program without any user input specifying selection of graphical program nodes and interconnection of graphical program nodes.*

Rather, as stated in the Abstract and col. 5, lines 32-41:

“During creation of the graphical program, the user operates to place an object node in the graphical program,...” and “This preferably includes the user arranging on the screen the graphical program, including the object node and various other nodes, and connecting the various nodes to create the graphical program” and “The user then configures the object node...”

Applicant further notes that column 5, lines 28-32 of Sojoodi 805 describes *manual* creation of a graphical program which is operable to access capabilities of an object, and specifically does *not* describe programmatically creating the graphical program via execution of a client program, *without any user input specifying the plurality of graphical program nodes or the interconnection of the plurality of graphical program nodes program during said creating.* Column 38, lines 55-59 describes manually adding items to the property node, where the properties node has been *manually* included (by the user) in a graphical program for accessing a software object. Column 6, lines 21-22 describes an application object and a manually included object node which is executable

to initiate execution of the application or to get/set properties of the application. Finally, column 6, lines 15-17 describes a software object that is an application or other re-usable software component. Nowhere does Sojoodi 805 teach or suggest that the object is a graphical program element or node.

In other words, Sojoodi 805 does not disclose programmatically creating the graphical program without user input specifying or guiding the inclusion and interconnection of graphical program nodes. More specifically, Sojoodi 805 does not disclose a *client-called server program*, which is executable to *programmatically create a plurality of graphical program nodes in the new graphical program, and programmatically interconnect the plurality of graphical program nodes in the new graphical program, without user input selecting or creating the nodes and specifying the interconnections between nodes.*

Rather, Sojoodi 805 describes *accessing* software objects that are provided by a server, but again, does *not* describe programmatically creating a graphical program without user input via execution of a client program (and/or server program). In fact, as shown above, Sojoodi 805 specifically describes creating a graphical program in response to user input, i.e., *not* programmatically as defined and described in the specification and new claims. In other words, Sojoodi 805 specifically does not teach executing a first graphical program to programmatically create a new graphical program without user input specifying or otherwise interactively participating in the creation process.

Additionally, as argued previously, the software objects of Sojoodi 805 are objects in an object-oriented or component sense, but are not properly graphical program objects or nodes, as described in the present application. For example, in column 4, line 54 – column 5 line 7, example objects include Active X objects, such as Microsoft Excel objects, Microsoft Access® objects, and Microsoft Word objects, OpenDoc® objects, and Common Object Request Broker Architecture (CORBA) objects. Further examples described include C++ and JAVA objects. Note that none of these software objects is a graphical program object or node as defined in the present application.

The Office Action cites Sojoodi 275 in an attempt to remedy the deficiencies of Sojoodi 805. However, Applicant notes that Sojoodi 275 also fails to teach this feature of

claim 285. Sojoodi 275 describes a system and method for creating a program for controlling an instrument independent of the interface type of the instrument, in a graphical program environment (Abstract). However, in Sojoodi 275, just as in Sojoodi 805, the creation of the program is performed *manually*, i.e., based on user input selecting and interconnecting graphical program nodes.

For example, as stated in col. 4, lines 41-53,

“The VISA nodes are advantageously *selected by a programmer* from a common VISA node palette...” and “Advantageously, *a programmer* is enabled to develop graphical programs to control VISA instruments...” and “...*the programmer* may develop graphical programs which comprise generic and specific VISA nodes for controlling an instrument...”

Furthermore, col. 5, lines 52-53 states:

“The VISA node icons in the block diagram comprise terminals which *the user wires together* using a wiring tool.”

Col. 6, lines 60-66 states:

“The method for controlling the instrument comprises displaying on the screen a VISA session icon which represents a VISA session to the instrument *in response to user input*, displaying on the screen a VISA function icon which represents performance of a VISA function in a VISA I/O control library on the instrument *in response to user input,...*”

Col. 7, lines 12-15 states:

“...the program editor displaying on the display screen the list of classes of the instrument, and *the user*

choosing a class from the list to associate with the instrument.”

In other words, just as in Sojoodi 805, in Sojoodi 275, the creation of the graphical program is specifically *not* performed programmatically *without user input selecting or creating the nodes and specifying the interconnections between nodes*, and additionally is *not* performed by a client/server system as claimed.

Applicant respectfully submits that neither Sojoodi 805 nor Sojoodi 275, either singly or in combination, teaches or suggests Applicant’s invention as represented in claim 285. Thus, for at least the reasons provided above, Applicant respectfully submits that claim 285, and claims dependent thereon, are patentably distinct and non-obvious over Sojoodi 805 and Sojoodi 275, and are thus allowable.

Claims 167-168, 170-205, 208-220, 222-241, 255-262, 264-271, and 284-285 were rejected under 35 U.S.C. 103(a) as being unpatentable over McDonald et al. (“McDonald”, USP 5,966,532), Sojoodi 275 and Sojoodi 805. Applicant notes that there was no claim 285 to reject, and assumes that the Examiner did not intend to include this claim number in the rejection. These claims have been cancelled, and so their rejection is rendered moot. Applicant believes that the new claims presented above are patentably distinct and non-obvious over McDonald, Sojoodi 805 and Sojoodi 275, and provides arguments detailing why below.

Applicant notes that in asserting motivation for the alleged combination of McDonald, Sojoodi 805, and Sojoodi 275, the Examiner has simply cited improved results (“the motivation being to enable the system to create the second graphical program from objects” and “the motivation being to enable the system to wire nodes to produce program logic and data flow”), which is improper. Applicant respectfully submits that neither McDonald, Sojoodi 805 nor Sojoodi 275 provides a motivation to combine, and so the references are not properly combinable to establish an obviousness rejection. Applicant further notes that even if Sojoodi and McDonald were combinable,

which Applicant argues they are not, the resulting combination would not teach Applicant's invention as claimed, as discussed in more detail below.

Moreover, Applicant also reminds the Examiner that, per *Interconnect Planning Corp. v. Feil*, 774 F.2d 1132, 1143, 227 USPQ 543, 551 (Fed. Cir. 1985) (it is insufficient to select from the prior art the separate components of the inventor's combination, using the blueprint supplied by the inventor). Applicant submits that the Examiner has selected particular portions of each of the cited references using Applicant's claim as a blueprint in an attempt to produce Applicant's claimed invention, which is improper.

New claim 285 recites:

285. A computer-implemented method for programmatically creating a graphical program, comprising:

creating a first program, wherein the first program is executable to programmatically create a new graphical program;

executing the first program, wherein said executing comprises programmatically creating the new graphical program, wherein said programmatically creating the new graphical program comprises:

programmatically creating a plurality of graphical program nodes in the new graphical program; and

programmatically interconnecting the plurality of graphical program nodes in the new graphical program;

wherein the interconnected plurality of graphical program nodes comprise at least a portion of the new graphical program; and

wherein said programmatically creating the new graphical program creates the new graphical program without any user input specifying the plurality of graphical program nodes or the interconnection of the plurality of graphical program nodes program during said creating.

The Office Action asserts that McDonald teaches a first program, when executing, programmatically creating a new graphical program, citing col. 3, lines 61-63, and further

asserts that while McDonald does not teach that the first program is a first graphical program and the creating a new graphical program by creating graphical program nodes and by interconnecting these created graphical program nodes, that these features are taught by Sojoodi 805, citing col. 5, lines 40-47. Applicant respectfully disagrees.

In the cited passage and elsewhere, Sojoodi 805 describes *the user* placing nodes in the graphical program, and *the user* interconnecting the nodes to create the graphical program. In other words, in Sojoodi 805, the creation of the graphical program is performed manually, and is specifically not performed programmatically via execution of a first graphical program, as claimed. Additionally, nowhere does Sojoodi 805 teach or suggest *wherein said programmatically creating the new graphical program creates the new graphical program without any user input specifying the plurality of graphical program nodes or the interconnection of the plurality of graphical program nodes program during said creating.*

Rather, as argued in the previous Response, McDonald teaches a wizard that selects a graphical code template in response to user selection of a user interface control, and includes the template in a graphical program, where the graphical code template corresponds to the selected control, and operates in conjunction with the control. In other words, the wizard selects a graphical code template in response to (and corresponding to) a user-selected user interface control, and inserts the template into the graphical program.

Nowhere does McDonald teach or suggest “creating a first program, wherein the first program is executable to programmatically create a new graphical program”, nor executing the first program to programmatically create the new graphical program, including *programmatically creating a plurality of graphical program nodes in the new graphical program, and programmatically interconnecting the plurality of graphical program nodes in the new graphical program...wherein said programmatically creating the new graphical program creates the new graphical program without any user input specifying the plurality of graphical program nodes or the interconnection of the plurality of graphical program nodes program during said creating.* The graphical code templates selected in McDonald have preconfigured nodes and interconnections, and hence McDonald does not perform, and has no reason to perform, any programmatic creation or interconnection of nodes.

As argued above, neither Sojoodi 805 nor Sojoodi 275 discloses or describes these features and limitations of claim 285, and so Sojoodi 805 and Sojoodi 275 fail to address the deficiencies of McDonald. Independent claims 333, 353, 355, 362, and 381 include similar limitations as claim 285 in that a program, e.g., graphical, program executes to programmatically create a graphical program *without any user input specifying selection of graphical program nodes and interconnection of graphical program nodes*. Means claim 379 includes the substantially similar limitations: wherein said means for programmatically instantiating the new graphical program are operable to *programmatically instantiate the new graphical program without any user input specifying instantiation of the new graphical program*; wherein said means for programmatically creating the node in the new graphical program are operable to *programmatically create the node in the new graphical program without any user input specifying creation of the node in the graphical program*; and wherein said means for programmatically invoking methods on the new graphical program or the node are operable to *programmatically invoke methods on the new graphical program or the node without any user input specifying invocation of methods on the new graphical program or the node*.

Thus, Applicant respectfully submits that the arguments presented above apply with equal force to these claims, and further submits that neither McDonald, Sojoodi 275, nor Sojoodi 805, either singly or in combination, teaches or suggests all of the features and limitations of claims 285, 333, 353, 355, 362, 379, and 381, and thus Applicant submits that these claims, and those claims respectively dependent thereon, are patentably distinct and non-obvious over the cited art, and are thus allowable for at least the reasons provided above.

Applicant also asserts that numerous ones of the dependent claims recite further distinctions over the cited art. However, since the independent claims have been shown to be patentably distinct, a further discussion of the dependent claims is not necessary at this time.

CONCLUSION

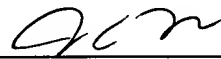
Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert & Goetzel PC Deposit Account No. 50-1505/5150-37301/JCH.

Also enclosed herewith are the following items:

- ☒ Return Receipt Postcard
- ☒ Request for Continued Examination

Respectfully submitted,



Jeffrey C. Hood
Reg. No. 35,198
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800
Date: 10/27/2004 JCH/MSW